COMP 4980 "Introduction to NLP"

Word-sense disambiguation

Stan Szpakowicz

Emeritus Professor

School of Electrical Engineering & Computer Science, University of Ottawa

TRU, Fall 2017

mostly Dan Jurafsky's material, edited by Stan Szpakowicz

The plan

1. Word senses and lexical relations
2. WSD in under an hour
3. That's it

. . . and WordNet

Word forms, lemmas, senses

A brief reminder

A *lemma* is a set of words with the same stem, the same major part of speech, and roughly the same sense — but possibly different affixes.*

A *word form* is essentially an element of a lemma — as it appear in texts.

A lemma can have more than one meaning. How many are possible?

Let us ask Guinness World Records.

Or Merriam-Webster.

* In English this is quite simple; only noun, pronoun and verb lemmas have more than one element.

Homonymy

Homonyms are words which share a form but have unrelated, distinct meanings.

Homographs are written the same.

- bass_1: stringed instrument, bass_2: fish
- bat_1: club for hitting a ball, bat_2: nocturnal flying mammal

Homophones are spoken the same.

- write and right
- piece and peace

Homonyms make trouble for many NLP applications such as, *e.g.,* information extraction ("bat care"?), machine translation ("bat" → "batte", or "bat" → "chauve-souris"?), or speech synthesis ("bass" → \'bās\, or "bass" → \'bas\?).

Polysemy

I withdrew money from the bank. [1]

The bank was constructed out of local red brick. [2]

Does bank in [1] and [2] have the same sense?

[1] A financial institution.

[2] The building belonging to a financial institution.

A polysemous word has *related* meanings.

Frequent words tend to have multiple meanings.

There exists systematic polysemy:
many words have the same type of multiple meanings.

Metonymy

One kind of systematic polysemy is called *metonymy*:

a figure of speech consisting of the use of the name of one thing for that of another of which it is an attribute or with which it is associated (such as "crown" in "lands belonging to the crown")*

Examples

The words school, university, hospital *etc.* can all mean the institution or the building. There is a systematic relationship:

Building ⇔ Owner of Building

Jane Austen wrote *Emma*. I love reading Jane Austen.

Author ⇔ Work of Author

The plum had beautiful blossoms. I ate a preserved plum.

Tree ⇔ Fruit of Tree

---

*with thanks to Merriam-Webster

More Greek words

How do we know that a word has more than one sense? Take the "zeugma test".

Zeugma is a figure of speech in which a word is related to two others in two different ways.* This is normally considered incorrect, or at least very odd.

"He is leaving for *greener pastures*."

"He is leaving for *ten days*."

? "He is leaving for *greener pastures* and *ten days*."

"He greeted them with *arms wide*."

"He greeted them with *expectations wide*."

? "He greeted them with *arms* and *expectations wide*."

*There is much more. Ask Wikipedia.

Synonymy

(That is a Greek word, too.)

Synonyms are words which have almost the same meaning, and can replace each other in most contexts:

automobile / car
big / large
couch / sofa
filbert / hazelnut
vomit / throw up
water / H_2O

No two words mean *exactly* the same, but the synonyms' meanings are close enough in most circumstances. The differences are often subtle, mainly stylistic, in formality, politeness, *etc.* Relevant concepts: genre and register.

Synonymy (2)

Synonymy is a relation between senses rather than words.

Consider the words big and large. This seems all right:

How big is that plane?

Would I be flying on a large or small plane?

This is not right:

Miss Nelson became a kind of big sister to Benjamin.

? *Miss Nelson became a kind of large sister to Benjamin.*

The difference explained:

big has a sense which means <u>older</u>, or <u>grown up</u>;

large lacks this sense.

Antonymy

Antonyms have senses opposite with respect to one feature of meaning. Otherwise, they are very similar: their other features are the same!

Categories of antonyms

- Complementary antonyms define a binary opposition:

  e.g., occupied / vacant, day / night, exit / entrance.

- Gradable antonyms are at the opposite ends of a scale:*

  e.g., dark / light, heavy / light, long / short, fast / slow.

- Relational antonyms are opposites in the context of a relationship:

  e.g., teach / learn, come / go, predator / prey, parent / child.

* or at a $\pm$ equal distance from the middle of the scale

Hyponymy and hypernymy

Word sense s_1 is a hyponym of sense s_2 if s_1 is more specific than, or denotes a subclass of, s_2:

car is a hyponym of vehicle; mango is a hyponym of fruit.

> Hypernymy is the inverse of hyponymy:
>
> vehicle is a hypernym of car; fruit is a hypernym of mango.

More formally, the class of things denoted by the hyponym is included in the class denoted by the hypernym.

Another view: sense s_1 is a hyponym of sense s_2 if being an s_1 necessarily means being an s_2.

Hyponymy and hypernymy (2)

Hyponymy is usually transitive:

> **if** being an s_1 means being an s_2
> **and** being an s_2 means being an s_3,
> **then** being an s_1 means being an s_3.

For example:

mango is fruit and fruit is produce, so mango is produce. In Artificial Intelligence one says that there is an IS-A hierarchy here.

But let us consider this example:

Kamloops is a city, a city is a municipality, therefore Kamloops is a municipality.

Kamloops, however, is a *specific* municipality.

So, IS-A can mean "an instance of" or "a subclass of".

Meronymy and holonymy

This is another Greek pair which name mutually inverse relations. They capture the concepts of *part* and *whole*.

There are several types of meronymy:

a genus has member meronyms subgenus and species;

a piece of pottery has a substance meronym clay;

a bicycle has part meronyms bicycle seat, bicycle wheel, chain, handlebar, pedal, and a few others.

In the same spirit:

a species has a member holonym genus;

clay has substance holonyms brick, tile and pottery;

a pedal has part holonyms organ, motor vehicle and bicycle.

WordNet

WordNet is a complicated resource (and it makes for a rather large topic). We will first look at it from far above, and then get our hands dirty by improvising in class.

Let us begin with statistics. WordNet 3.1, the latest version, dates back to ∼2007, so it is not new any more.[*] The last complete version was 3.0, © 2006. It is available for Unix systems, among others, thus for my Mac, but not for Windows.[†]

WordNet is a lexical database, a thesaurus and a kind-of dictionary. It is a net all right, but not of words: of *synsets*. A synset is a set of near-synonymous word senses which somehow make up a concept.

---

[*] Blame lack of funding. Seriously.

[†] Windows users get WordNet 2.1.

WordNet (2)

First off, we will ask WordNet about the word "set".

Right. Now, let us revisit the WordNet Web interface and play with the word "chump".

Nine word senses share the underlying concept. The words are chump, fool, gull, mark, patsy, fall guy, sucker, soft touch, mug. Of these, mark is highly polysemous, while gull and mug have meanings unrelated to naïveté.

Next: the hypernym hierarchy for mug^4 and gull^2.* Note the deeper hierarchy for the biological term.

Such fun has no end, really. ☺

* That is called "inherited hypernym"

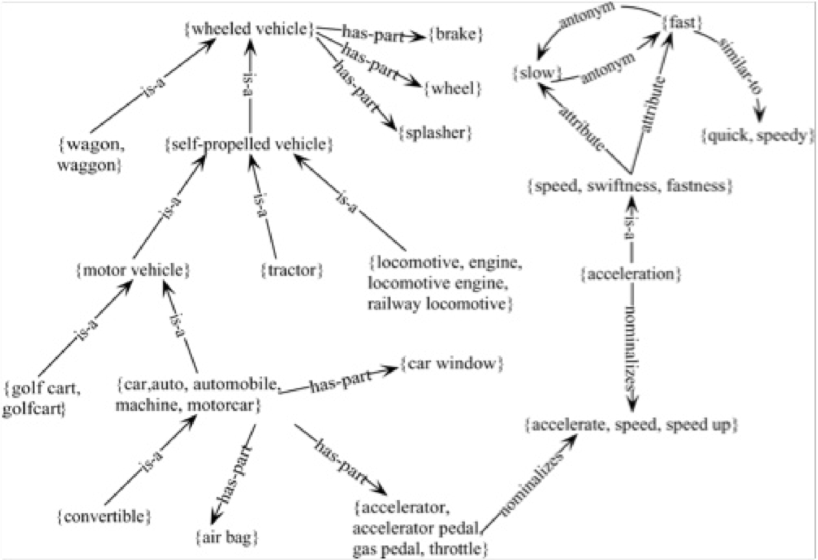
WordNet noun relations

Relation	Also Called	Definition	Example
Hypernym	Superordinate	From concepts to superordinates	$\textit{breakfast}^1 \rightarrow \textit{meal}^1$
Hyponym	Subordinate	From concepts to subtypes	$\textit{meal}^1 \rightarrow \textit{lunch}^1$
Instance Hypernym	Instance	From instances to their concepts	$\textit{Austen}^1 \rightarrow \textit{author}^1$
Instance Hyponym	Has-Instance	From concepts to concept instances	$\textit{composer}^1 \rightarrow \textit{Bach}^1$
Member Meronym	Has-Member	From groups to their members	$\textit{faculty}^2 \rightarrow \textit{professor}^1$
Member Holonym	Member-Of	From members to their groups	$\textit{copilot}^1 \rightarrow \textit{crew}^1$
Part Meronym	Has-Part	From wholes to parts	$\textit{table}^2 \rightarrow \textit{leg}^3$
Part Holonym	Part-Of	From parts to wholes	$\textit{course}^7 \rightarrow \textit{meal}^1$
Substance Meronym		From substances to their subparts	$\textit{water}^1 \rightarrow \textit{oxygen}^1$
Substance Holonym		From parts of substances to wholes	$\textit{gin}^1 \rightarrow \textit{martini}^1$
Antonym		Semantic opposition between lemmas	$\textit{leader}^1 \Longleftrightarrow \textit{follower}^1$
Derivationally Related Form		Lemmas w/same morphological root	$\textit{destruction}^1 \Longleftrightarrow \textit{destroy}^1$

WordNet verb relations

Relation	**Definition**	**Example**
Hypernym	From events to superordinate events	*fly*^{9} $\rightarrow$ *travel*^{5}
Troponym	From events to subordinate event (often via specific manner)	*walk*^{1} $\rightarrow$ *stroll*^{1}
Entails	From verbs (events) to the verbs (events) they entail	*snore*^{1} $\rightarrow$ *sleep*^{1}
Antonym	Semantic opposition between lemmas	*increase*^{1} $\Longleftrightarrow$ *decrease*^{1}
Derivationally Related Form	Lemmas with same morphological root	*destroy*^{1} $\Longleftrightarrow$ *destruction*^{1}

WordNet as a knowledge base



The plan

1. Word senses and lexical relations
2. WSD in under an hour
3. That's it

A glance at a few simple methods

Resolving ambiguity

To resolve sense ambiguity — polysemy, homonymy and so on — we need context:

- text statistics,
- dictionaries and thesauri.

The upper bound is human performance in disambiguation. The lower bound can be established by always selecting the most frequent sense. This baseline still requires reliable sense statistics.

For example, 80% disambiguation accuracy is a good result for two equally probable senses: 30% over chance. On the other hand, 80% success rate on two senses with the 80-20 distribution would be no better than chance.

WSD, procedures

Types of disambiguation procedures:

- unsupervised,
- supervised,
- dictionary-based,
- thesaurus-based,
- graph-based.

Unsupervised WSD can be, *e.g.*, based on corpus analysis. It is worthwhile but not easy to explain briefly. That it why we will not discuss unsupervised methods here.

You might be interested in details. Other methods depend on resources which may not always be easy to get, or may be less than adequate for the purpose. If you are interested, see Ted Pedersen's book chapter for a somewhat technical overview.

Supervised WSD

The data: a sense-annotated corpus for training.

The method: supervised machine learning.

The result: disambiguation rules (we must evaluate their quality).

The procedure will be described for a single word w with k senses $w_1, \ldots, w_k$.

We determine all contexts of w in the corpus: $c_1, \ldots, c_n$. We also need the corpus vocabulary: words $v_1, \ldots, v_p$ which appear in contexts for w (p may be rather large).

We want sense w_m which maximizes conditional probability $P(w_j|c)$ of a word sense given a context:

$$\text{for all } j \neq m,\ P(w_j|c) \leq P(w_m|c)$$

Supervised WSD (2)

We rely on an old acquaintance — Bayes's theorem:

$$P(w_j|c) = P(c|w_j) * P(w_j)/P(c)$$

Let $c = v_a v_b \ldots v_y v_z$. The "naïve Bayes" assumption states that context elements are independent. So:

$$P(c|w_j) \approx P(v_a|w_j) * P(v_b|w_j) * \ldots * P(v_y|w_j) * P(v_z|w_j)$$

The MLE probability $P(v_i|w_j)$ comes, as usual, from counting co-occurrences of v_i and w_j in the corpus. The same goes for w_j itself.

$$P(v_i|w_j) = count(v_i, w_j)/count(w_j) \quad [1]$$

$$P(w_j) = count(w_j)/count(w) \quad [2]$$

Supervised WSD (3)

$P(c)$ is constant: maximization of $P(w_j|c)$ ignores it.

Also, we can maximize the logarithm of a value instead of maximizing the value itself.

So, we will find w_m which maximizes

$$P(v_a|w_j) * \ldots * P(v_z|w_j) * P(w_j)$$

when we find w_m which maximizes

$$log\ P(v_a|w_j) + \ldots + log\ P(v_z|w_j) + log\ P(w_j) \qquad [3]$$

The next slide shows the algorithm of supervised WSD.

Supervised WSD (4)

Step 1 (training)

- Calculate [1] for all context/word-sense pairs $v_i\ w_j$.
- Calculate [2] for all senses w_j.

Step 2 (testing)

- Given word w, let w_m be w_j which maximizes [3].

As an example, take context elements which help disambiguate the word *bank*:

- the financial sense is likely to co-occur with *interest*, *teller*, *account*, . . .
- the river-side sense can be signalled by such clues as *water*, *river*, *right*, . . .

Dictionary-based WSD

This method assumes that a good dictionary is available. (WordNet qualifies as a dictionary, even if non-standard.)

Given a word w, we need a dictionary definition D_{w_k} for every sense w_k of that word.

We represent this definition as a bag of words (an unordered collection with repetitions).

In particular, we have a dictionary definition $D_{v_{ij}}$ for the j^{th} sense of every vocabulary word v_i. Let E_{v_i} be the union of all $D_{v_{ij}}$, also represented as a bag of words.

Michael Lesk proposed in 1986 a simple disambiguation algorithm based on dictionary definitions.*

*Here is the original paper.

Dictionary-based WSD (2)

Here is one of several versions of Lesk's algorithm.*

- Take word w and its context $c = v_a v_b \ldots v_y v_z$.
- Calculate the union $D_c = E_{v_a} \cup E_{v_b} \ldots E_{v_y} \cup E_{v_z}$.
  It is a dictionary "replica" of the meanings of the context words.
- Score every w_k by the overlap between D_{w_k} and D_c.
  Overlap is measured in words (usually stemmed) common to the two bags of words.
- Select the maximizing w_k.

The algorithm is not very accurate. Many people have tried to improve it. You might want to *wikipedia* this matter.

NLTK has a Lesk method. It is not terribly impressive.

*There even is "corpus Lesk".

Thesaurus-based WSD

We assume that every word sense w_k has a distinguishing semantic tag $t(w_k)$. A tag can be:

- a WordNet synset number,
- a sense number in the *Merriam-Webster* dictionary,
- a path in a thesaurus,
- and so on.

Now, we define the distance d between a semantic tag t and a word v it describes:

$d(t, v) = 1$ if t is v's tag $\quad\quad d(t, v) = 0$ otherwise

This kind of binary distance has the advantage of being easily calculated, and it does not unduly favour any words.

Thesaurus-based WSD (2)

Here is a disambiguation algorithm which avails itself of semantic tags.

- Take word w and its context $c = v_a \ldots v_z$.
- Score every sense w_k by the value of $d(t(w_k), v_a) + \ldots + d(t(w_k), v_z)$.
- Select the w_k which maximizes the score.

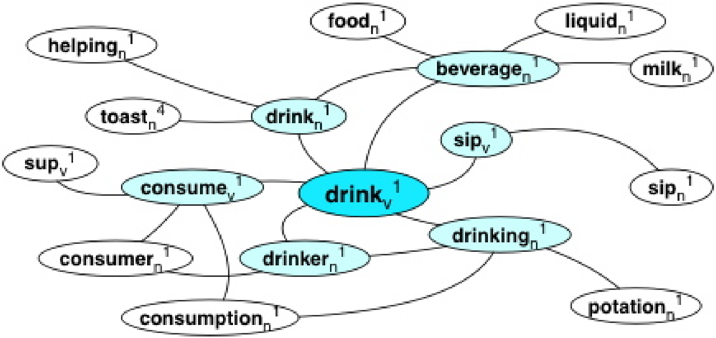
This algorithm is only as good as the thesaurus it works with.

For example, it tends to miss proper nouns.

That is a pity. Proper nouns make good context. Sadly, they are poorly represented in thesauri and dictionaries.

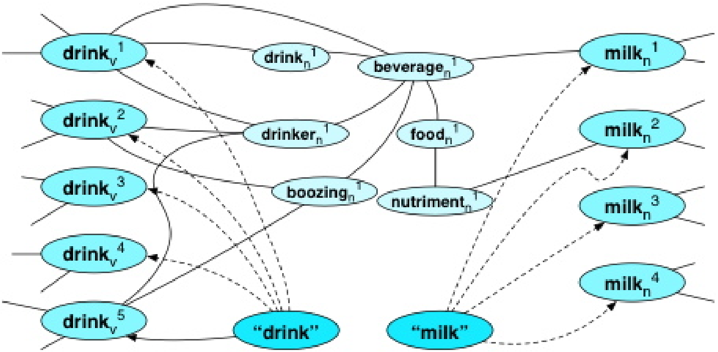
Graph-based WSD

WordNet can be seen as a graph with word senses as nodes, and relations — hypernymy and meronymy — as edges. We add some edges between word senses and words in glosses.



Graph-based WSD (2)

Example: "she drank some milk". We connect "drink" and "milk" to the WordNet graph, and select their *most central* senses according to a measure inspired by PageRank.*



*No more will be said here. ☹

Evaluation

. . . will not be touched here ☹

Sorry. But remember that no NLP system can be trusted unless it has been evaluated.

The plan

That's it

December is near.

It will soon be over. ☺